

[INICIAR SESIÓN](#)[NUESTROS PLANES](#)[TODOS LOS CURSOS](#)[FORMACIONES](#)[CURSOS](#)[PARA EMPRESAS](#)[ARTÍCULOS DE TECNOLOGÍA > FRONT END](#)

Virtual DOM



Nathalia Cardozo

29 de Marzo

La expresión “manipular el DOM” es muy común y muy escuchada cuando empezamos a estudiar desarrollo web. Este artículo abordará el concepto de lo qué es el Virtual DOM, sus diferencias con el DOM real y cuándo debemos usarlos. Si aún tienes dudas sobre cuál es el DOM real, tenemos un artículo que explica el tema en detalle: “[¿Qué es DOM?](#)”.



¿Qué es Virtual DOM?

Ahora bien, vamos directo al punto. ¿Qué es Virtual DOM? ¿Y cuándo debemos utilizarlo? Si consideramos su nombre, podemos inferir que se trata de un documento virtual. Pero ¿qué queremos decir con eso?

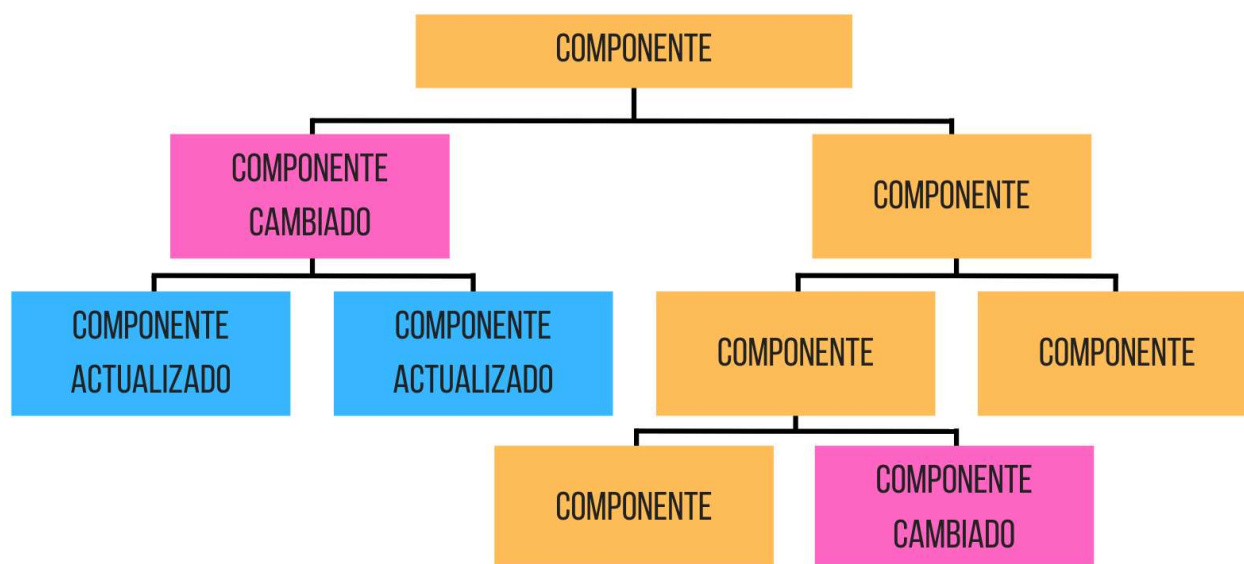
El Virtual DOM es una representación visual del DOM, así que no es el DOM real. Esta representación se almacena en la memoria del navegador y se sincroniza con el DOM real a través de una biblioteca como el React DOM, por ejemplo. En el Virtual DOM, tenemos un estado intermedio entre lo que vemos en pantalla y lo que tenemos en el DOM real.

Cuando se aplica una alteración a través de una biblioteca, el Virtual DOM guarda esta nueva versión del DOM en la memoria del navegador, solo con los componentes modificados, independientemente de su posición en el árbol DOM.

Imagina, por ejemplo, que nuestra página y sus componentes se parezcan exactamente como un árbol real y que, a cada cambio de estación, este árbol sufra cambios en su estructura. Recuerdas que en un árbol tenemos las raíces, el tronco, las ramas y las hojas, así como los componentes de nuestra página.

Si el comportamiento de nuestro árbol fuera como el comportamiento del DOM real, cada vez que la estación del año se cambia de verano a otoño, por ejemplo, en lugar de solo caer las hojas, todo el árbol cambie por completo, incluso las partes que no sufrieron cambios, como el tronco y las ramas, por ejemplo. La situación es como se muestra en la siguiente imagen.

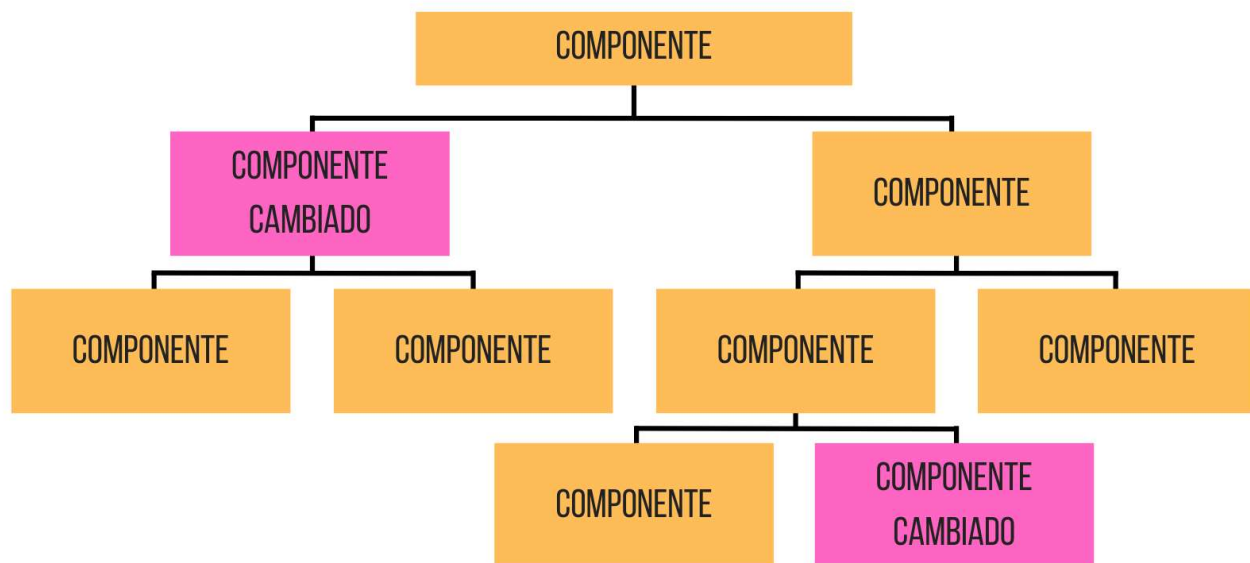
Árbol de componentes del DOM



En la imagen, presentamos un árbol de componentes basado en el DOM real, donde dos componentes han sufrido cambios (componente cambiado) fueron actualizados, pero sus componentes hijos que no han sufrido cambio (componente actualizado) también se han actualizado.

Por otro lado, si el comportamiento de este árbol fuera como el comportamiento del Virtual DOM, tendríamos algo mucho más parecido a lo que ocurre en la vida real. Con el inicio de una nueva estación, solo se ven afectados los componentes que cambian, así que, el tronco, las raíces y las ramas quedarían con el estado actual y solo las hojas sufrirían esta alteración, como podemos ver en la imagen de abajo.

Árbol de componentes del Virtual DOM



En la imagen solo se actualizaron los componentes que sufrieron cambios (componente cambiado), sin que sus hijos sean afectados.

Sin embargo, es necesario recordar que el DOM virtual es solo una representación visual y que el DOM real todavía existe y ahí es donde necesitamos tener la información al final. Aquí es donde entra el trabajo más importante de nuestra biblioteca, que es sincronizar lo que vemos en el DOM virtual con lo que tenemos en el DOM real.

Cuando la aplicación está sincronizada, lo que sucede automáticamente en la biblioteca en uso, la versión que el DOM virtual tiene en memoria se replicará en el DOM real, que a su vez funciona para actualizar también cualquier componente hijo que pueda existir en la estructura de la aplicación.

¿Cuáles son las ventajas y diferencias del Virtual DOM con el DOM real?

Cambiar cualquier componente o elemento directamente en el DOM real puede resultar muy costoso para el proceso de desarrollo. Sin embargo, es inevitable cambiar el DOM cuando necesitamos que se muestre alguna modificación en la aplicación que estamos

utilizando. Así, Virtual DOM viene a cubrir esa necesidad de realizar cambios que parecen casi instantáneos pero que no comprometen el rendimiento de la aplicación.

Al cambiar el DOM real, el navegador debe reprocesar su contenido para que se apliquen los cambios, y lo hace recargando la página de la aplicación. Mientras que con Virtual DOM, esta necesidad no existe, porque lo que vemos en pantalla no es el DOM real, sino su representación visual.

O sea, volviendo a nuestra analogía del árbol, si hiciéramos cambios directamente en el DOM real, nuestro árbol cambiaría (o se actualizaría) por completo y este proceso, en comparación con el proceso del Virtual DOM, es mucho más laborioso que cambiar solo lo que debe cambiar.

Esta tecnología nos permite, entonces, crear y ejecutar con mayor facilidad aplicaciones denominadas **SPA (o Single Page Application)**, donde tenemos numerosos componentes que se actualizan simultáneamente sin necesidad de que la página se renderice con cada alteración, ya que solo los componentes que fueron cambiados se actualizarán.

Además, para utilizar el DOM Virtual es necesario utilizar alguna biblioteca que intermedia nuestro DOM real con el virtual. También es necesario entender que, aunque estemos utilizando el DOM Virtual en nuestra aplicación, el DOM real no dejará de existir, sino que la comunicación con él se realizará de forma automatizada y cualquier interacción que realice el usuario no tiene un efecto inmediato en el DOM real, solo en el virtual.

Conclusión

En el desarrollo Front-End, es inevitable trabajar con manipulación DOM y esta tarea puede ser bastante costosa a veces. Sin embargo, el Virtual DOM viene a ayudar de varias formas en el desarrollo Front-End, permitiendo la creación de arquitecturas más complejas permitiendo la creación de arquitecturas más complejas así como una mejor gestión del estado por parte de la aplicación.



Nathalia Cardozo

¡Hola! Soy Nathalia y soy estudiante de Análisis y Desarrollo de Sistemas y también soy parte del equipo de Scubas de Alura Latam. Además, también soy desarrolladora Front End

con enfoque en ReactJS.

Referencias

[JavaScript HTML DOM](#)

[Introducción al DOM](#)

[DOM virtual y detalles de implementación](#)

Cursos de Front End

ARTÍCULOS DE TECNOLOGÍA > FRONT END

**En Alura encontrarás variados cursos sobre Front End.
¡Comienza ahora!**

SEMESTRAL

US\$49,90

un solo pago de US\$49,90

- ✓ 218 cursos
- ✓ Videos y actividades 100% en Español
- ✓ Certificado de participación
- ✓ Estudia las 24 horas, los 7 días de la semana

- ✓ Foro y comunidad exclusiva para resolver tus dudas
- ✓ Acceso a todo el contenido de la plataforma por 6 meses

¡QUIERO EMPEZAR A ESTUDIAR!

Paga en moneda local en los siguientes países

ANUAL

US\$79,90

un solo pago de US\$79,90

- ✓ 218 cursos
- ✓ Videos y actividades 100% en Español
- ✓ Certificado de participación
- ✓ Estudia las 24 horas, los 7 días de la semana
- ✓ Foro y comunidad exclusiva para resolver tus dudas



Acceso a todo el contenido de la
plataforma por 12 meses

¡QUIERO EMPEZAR A ESTUDIAR!

[Paga en moneda local en los siguientes países](#)

Acceso a todos
los cursos

Estudia las 24 horas,
dónde y cuándo quieras

Nuevos cursos
cada semana

NAVEGACIÓN

PLANES
INSTRUCTORES
BLOG
POLÍTICA DE PRIVACIDAD
TÉRMINOS DE USO
SOBRE NOSOTROS
PREGUNTAS FRECUENTES

¡CONTÁCTANOS!

¡QUIERO ENTRAR EN CONTACTO!

BLOG

PROGRAMACIÓN
FRONT END
DATA SCIENCE
INNOVACIÓN Y GESTIÓN
DEVOPS

AOVS Sistemas de Informática S.A
CNPJ 05.555.382/0001-33

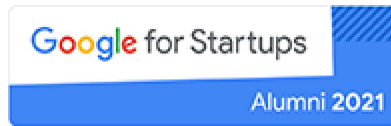
SÍGUENOS EN NUESTRAS REDES SOCIALES



ALIADOS



En Alura somos unas de las Scale-Ups seleccionadas por Endeavor, programa de aceleración de las empresas que más crecen en el país.



Fuimos unas de las 7 startups seleccionadas por Google For Startups en participar del programa Growth Academy en 2021

POWERED BY

CURSOS

Cursos de Programación

Lógica de Programación | Java

Cursos de Front End

HTML y CSS | JavaScript | React

Cursos de Data Science

Data Science | Machine Learning | Excel | Base de Datos | Data Visualization | Estadística

Cursos de DevOps

Docker | Linux

Cursos de Innovación y Gestión

Productividad y Calidad de Vida | Transformación Ágil | Marketing Analytics | Liderazgo y Gestión de Equipos | Startups y Emprendimiento